

REMARKS/ARGUMENTS

The Office Action dated May 21, 2007 has been carefully reviewed and considered. Claims 1-15, 28, 29, and 37 are currently pending and Claims 1-15, 28-29 and 37 are rejected.

Claim 3 is rejected under 35 U.S.C 112 as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regard as the invention. Claim 1-3, 5-6, 9-15, and 37 are rejected under 35 U.S.C. 102(e) as being anticipated by Anderson et al. (U.S. 2003/0084128 AL). Claim 4 and 7-8 are rejected under U.S.C. 103(a) are being un-patentable over Anderson et al. (hereinafter Anderson) (U.S. 2003/0084128 AL) in view of Zhao (U.S 6,035,404). Claim 28-29 are rejected under 35 U.S.C. 103(a) are being un-patentable over Anderson et al. (hereinafter Anderson) (U.S. 2003/0084128 AL). in view of Jackson et al. (hereinafter Jackson) (U.S 2002/0152305 AL).

In the foregoing Amendments, Claims 3, 37, and other multiple claims have been amended. Support for these amendments can be found in the specification and claims of the application as filed. No new amendment has been made in specification. No new matter has been added. Claims 30 has been canceled without prejudice or disclaimer of the subject matter contained therein. Claims 1-15, 28-29 and 37 are currently pending.

Applicant respectfully requests entry of the foregoing Amendments and reconsideration of the present application in light of the amendments above and the remarks below.

Claim Objections

Claims 37 is objected to as allegedly being improper formalities and so forth. Claim 37 has been amended and this objection is traversed. Claim 30 is objected due to the Election/Restriction rule and Claim 30 was restricted into Group II. Claim 30 is canceled and this objection is traversed.

Objection to Specification under 35 U.S.C. 132 (a)

The Office Action objects the disclosure because there are couples of new matter throughout the Specifications. . Now the original materials of line 11, page 15 becomes line 4 of page 12, and also line 13, page 15 of original material becomes line 6 of page 12

in present version. The new matter has been canceled and this objection is traversed. There is an additional amendment on page 7, line 23-24 of this version.

Rejection under 35 USC 112

Claim 3 has been amended, the indefinite has been removed, and this objection is traversed.

Rejection under 35 USC 102(e)

The paragraph 5 of Office Action has rejected claims 1-3, 5-6, 9-15, and 37 under Anderson. Without admitting that Anderson is prior art and reserving the right to establish that it is not prior art, Applicant respectfully traverses the rejections for the reasons below:

Regarding independent Claims 1, the Office Action asserts that Anderson teaches “a method for supporting multiple simultaneous concurrent tasks within a single web-console in a central controlled distributed scalable virtual machine (“CCDSVM”) environment, said method comprises

(a) a user logging in from a web-console of a console host to said CCDSVM environment ([0037],[0072]; (b) said user from said web-console of said console host obtaining all information of the target system within said CCDSVM environment (from task queue 14) ([0050],[0082]); (c) said user from said web-console on said console host can select any available target system and initiate multiple concurrent tasks without delay on targeted system based on said all obtained information of said CCDSVM environment (from task queue 14) (0032-0034),[0050]); (d) console supporting software on control management station getting and storing tasks into a user space task list, and obtaining associated locks for each task([0032],[0040],[0058],[0059]; and (e) console support software distributing multiple tasks to multiple systems until all of said tasks are performed. ([0082]-[0083]) “

The Applicant disagrees with such assertion of the Office Action for the reasons below.

“A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference” Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628,631,2USPQ2d 1051, 1053(Fed. Cir.1987). “The identical invention must be shown in as complete details as is contained in the claim.” Richardson v Suzuki Motor Co., 869F.2d 1226,1236, 9USPQ2d 1913, 1920 (Fed. Cir. 1989) (See also MP.E.P 2131). Here and every element of Applicant’s independent claims 1 and are not found, either expressly or inherently, in a single prior art reference. Applicant respectfully submits that Anderson can not anticipate the presently claimed invention because Anderson fails to disclose each and every element set forth in the claimed invention either inherently or expressly.

For example, it is easy to prove that Anderson has failed to suggest for supporting user to perform multiple concurrent simultaneous tasks from a single web-browser.

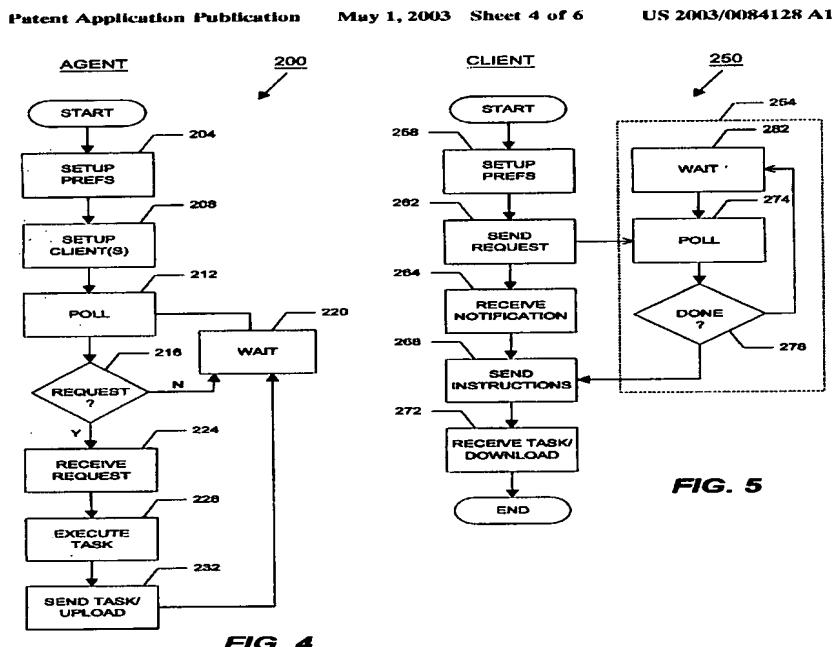
A user performs multiple concurrent tasks from a single web-browser UI means that at a given period of time said user is able to initiate and run at least two tasks from a same web-browser UI, and said two tasks will run in parallel at least at some point of time during that given period of time.

In order for a user successful in initiating simultaneous multiple concurrent tasks within a same web-browser UI, a important condition must be met is that each task initiated by such user must not be blocked at same web-browser UI, so that such user can immediately initiating another task after previously initiated task without wait in same web-browser. This is similar to the condition for supporting multi-tasks in command line based or native window based user operating environment.

Anderson’s system has immediately violated this condition because with Anderson’s system user has to wait, and even worse, user can logs off server after initiating a task, then periodically pull and check server to see if the initiated task has been finished. This has been revealed in Anderson’s [0077] that

“Once the task request is sent to the server 10, the remote client 20 will wait for a reply from the server 10. According to one embodiment, the remote client 20 logs off of the server 10 and the polls the server 10 periodically to determine whether the task request was completed by the local agent 40....” ([0077] of Anderson)(Emphasis Added).

In addition, Anderson's Fig. 5 has revealed same problem. For example, with optional/alternate act 254, where user enters waiting and polling state. Third, let us look at same Fig. 5 of Anderson between act 262 and 264, where a user also has to wait for receiving a notification or signal, or wait in act 250.



The results of the waiting and pulling is that the user of Anderson can not initiating another concurrent task immediately following the previously initiated task from same web-browser. In other words, user of Anderson has failed to initiate multiple simultaneous concurrent tasks with a same single web-browser.

Fig. 5 of Anderson demonstrates the task executing path of Anderson. [0077] and Fig. 5 of Anderson has clearly reveals that even Anderson provides user a task queue disclosed in [0050], [0082], [0032]-[0034] of Anderson, user of Anderson still has been blocked and has to wait either in pulling state or wait for notification from server. The one with ordinary skill in art can understand that if a user initiating a task from a web-browser UI and has to wait for task's completion or just log off server then pull the server

for task's completion, it will be impossible to let such user to initiating another task from same web-browser UI and let them run concurrently. Therefore, these two tasks will never be run in parallel at least at some point of time during a given period of time. It is clearly that, Anderson's system can not support multiple concurrent tasks with a single web-browser, but only support to run task one after another without concurrency.

With Anderson, for example, to get 2-GB or even 500MB file on Internet, or configure a 1TB raid controller, or other time consuming tasks with Anderson's method, in either cases of waiting or logging off server, the user of Anderson will have no chance to use the same web-browser on their hand to perform other tasks concurrently and user's web-screen also will experience a effect of web-browser screen-freezing and can do nothing during this waiting period. Even if the user of Anderson performs downloading a few bytes of small file and user do not feel the web-screen freezing due to tasks' short time of execution, the user of Anderson, as matter of the fact, has to wait the completing the task of downloading few bytes, then initiating another task. This also violates the principle of task concurrency in computer industry's conventional term that at a certain giving period of time two or more tasks can be run parallel at least at some point of time without waiting for each other task's completion.

Obviously, Anderson has failed to suggest or disclose concurrent multi-tasks support within a single web-browser UI because his invention only can work for very simple and non time consuming tasks, which only need to be run one after another and do not require to be run concurrently.

On the other hand, the present invention successfully provides a user from a single web-browser to initiate multiple concurrent tasks on systems in CCDSVM environment due to two facts: one is that the present invention providing user space task list to avoid tasks being blocked or waited, and to protect data integrity of each tasks. Another is that the present invention has provided a way to let user do not have to wait for the completion of the tasks being initiated within same web-browser UI on web-console, therefore, user can immediately start initiating another task from same web-browser UI.

Lines 1 - 11 on page 14 of present invention has reveals such points that

“The credential of executing a specific task on a specific target system by user A is checked. In addition, the ordinary users access & operation permissions and credentials are setup by administrator with super user privileges. If user A is not permitted to perform any task on such target system or is not permitted to perform such task on any system, the task execution will be fail and user A will be notified via net (11 or 12 of Fig. 2). Otherwise, the task will be carried out by the corresponding thread on target system, which is either control management station (2 of Fig. 2) or a system unit (3 of Fig. 2). If there is needs, the console supporting software (6 of Fig. 2) will send results data back to web-console (9 of Fig. 2). When a task is either failed or succeeded, the threads of console supporting software (6 of Fig. 2) will release the locks acquired for this task.”

We shall notice that with present invention, only “If there is needs, the console supporting software (6 of Fig. 2) will send results data back to web-console”, in other words, normally the user does not have to wait for the results or notification at their web browser UI . This adequately ensures that user can initiate other tasks immediately follow previous task from same single web UI of present invention and without waiting for previous task to be complete. Therefore, multiple tasks could be run concurrently and parallel at some point of time in a given period of time.

With the support of multiple users concurrent login into said CCDSVM of present invention, multiple users can concurrently login CCDSVM and each user can perform multiple concurrent tasks from each of their own single web-browser over one or multiple systems in CCDSVM.

Clearly, present invention has overcome a major hurdle that Anderson has encountered and failed to suggest. Because Anderson’s method has failed to support user initiating multiple concurrent tasks within same web-browser, Anderson has failed to suggest the major elements and the limits of present invention. Consequently, the independent Claim 1 is patentable over Anderson and the rejection of Office Action on Claim 1 of present invention shall be withdrawn.

For another example, the Claim 1, bullet (a), (b), and (c) of present invention recites “(a) a user logging in from a web-console of a console host to said CCDSVM

environment”, “(b) said user from said web-console of said console host obtaining all information of the targeted systems within said CCSDVM environment”, and “(c) said user from said web-console of said console host selecting a target system and initiating tasks based on said all information of said CCDSVM environment”. (Emphasis added)

There are other three major points that Anderson has failed to suggest in his invention: First, users of Anderson have no ability to obtain all information of the targeted systems within CCDSVM environment; Second, users of Anderson have no ability of selecting any target system from multiple targeted systems and initiating tasks based on said all information of said CCDSVM environment; Third, Anderson’s system can not perform the tasks in range of present invention. Let us explore why Anderson has failed to suggest each element and limit of present invention in this area.

First, the [0032]-[0034], [0050], [0082] of Anderson only disclose that how the users’ tasks can be queued in task queues, how the local agent of home system can periodically detect if there is task for it, and how the remote client can initiating a task and wait for completion notification or logs off and periodically pull if the task is completed.

As mater of the fact that the remote client system of Anderson is manually pre-configured into agent software to establish a 1-to-1 based configuration, so that each user can exclusively access only one local system, which typically located at home. This has been revealed in [0071] and [0079] of Anderson:

“[0071] In act 208, remote clients 20 can be setup. This can be done manually, by configuring the remote client in the agent software 200, or it can be done automatically. What is meant here is that remote clients 20 can be setup and managed, thereby giving a user of the local agent software 200 the ability to individually tailor access, security, or file transfer type information for particular remote clients, or globally setting such preferences, with respect to a the local computer the local agent is associated with.” (Emphasis added) “[0079] FIG. 6 is a flowchart for the server software 300. We begin with act 304, where the relational database management system 16 is setup. Here, we can setup remote client 20 and local agent 40 default values, such as polling period for the local agent 40, file types for different remote client 20 types, notification messages, upload file types, and other standard information concerned with the file management. Preferably, pricing plans and other user information is stored in the database 16, so it can be setup too. In act 308, users are setup for the database 16. This can be via manual entry, or an automated process that is part of a HTML or XML based web interface on the server 10. We note that an exemplary database

schema for the database 16 is depicted in FIG. 7 and described below with reference that figure.” (Emphasis added)

The Anderson’s [0071] and [0079] clearly reveals that his remote client can only access one fixed system because the client is manually or automatically configured with a fixed local agent. As mater of the facts that the entire specification of Anderson has never mention that a user at remote client system can obtain all information of targeted systems within Anderson’s remote file access system environment. Also, Anderson never mentions that a client at a single web-browser can select a target system from multiple local systems, and further can not to initiate multiple concurrent tasks based on said all information of Anderson’s file access system. Another significant difference is that in the present invention, the target system, on which the task will be initiated, could be control management station or any system unit. On the other hand, with Anderson, user of remote client only can log on server of Anderson and initiate task on a fixed local system but not initiate task on server. For example, the present invention describes that:

“The credential of executing a specific task on a specific target system by user A is checked. In addition, the ordinary users access & operation permissions and credentials are setup by administrator with super user privileges. If user A is not permitted to perform any task on such target system or is not permitted to perform such task on any system, the task execution will be fail and user A will be notified via net (11 or 12 of Fig. 2). Otherwise, the task will be carried out by the corresponding thread on target system, which is either control management station (2 of Fig. 2) or a system unit (3 of Fig. 2). If there is need, the console supporting software (6 of Fig. 2) will send results data back to web-console (9 of Fig. 2). When a task is either failed or succeeded, the threads of console supporting software (6 of Fig. 2) will release the locks acquired for this task.” (paragraph 4 of page 10 of present invention)

Third, the Anderson’s invention can not perform the tasks in claim 1 of present inventions, which has been fully implemented at the time these claim were written, such as described (in part) in claim 14 and page 15 of specification that:

“a) Move or transmit data such as a multiple gigabytes of file or other data in any form from any point or any system to another point or system within CCDSVM (Fig. 2).

- b) Configure, partition and design entire storage system (raid/disk) within CCDSVM
- d) Monitor and display activities and status for network, storage, CPU, processes and threads...”.

These tasks never being disclosed by Anderson and can not be performed by Anderson's invention.

In summary, Anderson has failed to disclose a wide range of tasks of present invention, failed to disclose to let user obtain all information from group of systems of present invention, failed to disclose to let user having choices of selecting any target system from all systems of present invention, failed to disclose to let user initiating tasks based on selected target system and based on obtained all information of group of system of present invention, and more significantly failed to disclose to let user initiating multiple concurrent tasks within a same web-browser UI of present invention. Consequently, Anderson has failed to suggest and anticipate the elements of present invention either expressly or inherently. Applicant submits that the Claim 1 is fully patentable over Anderson's invention. The rejection of the Office Action on Claim 1 must be withdrawn.

Regarding independent claim 37, the Office Action has rejected claim 37 under 35 U.S.C. 102, alleging that “Anderson teaches a method for supporting multiple simultaneous concurrent tasks with a single web-consoles comprises:

Providing a group of computer systems having at least one control system and zero or more server systems and connected together through network media, wherein said group of computer systems are controlled, operated, and managed by said control system with a set of software modules running on either control system or on each server systems in said group of systems (Fig. 1, item 4, 10, 20, 40,44, 48, etc.);

Providing multiple users login concurrently each from web-browser of client system into said control system ([0037],[0072]);

Providing said users each from a single web-browser on client system to obtain information relating to system configuration and resources information

of control system and server systems within said group of systems([0050],[0082]);

Providing said users each from a single web-browser of client systems to select said target systems, which is either a said control system or server systems, and to initiate multiple simultaneous concurrent tasks over the said configuration and resources information on selected target systems ([0082]-[0083]);

Providing said web console supporting software on control management station gets and stores tasks from each users on client systems into an user space task list, and also obtains the associated locks for each tasks ([0032],[0040],[0058],[0059], Fig. 1, item 14); and

Executing tasks arranged by said console supporting software, which run on the target systems until the tasks got finished ([0032],[0040],[0058],[0059]). Anderson describes how a single client or computer interacts with a web server, it also describes that there are multiple computers or clients on the network that can simultaneously access the internet web sever ([0029]-[0030]. etc.).”

In support of such assertion, the Office Action cites Anderson ([0037],[0072], ([0050],[0082]), ([0082]-[0083]), and ([0032],[0040],[0058],[0059]). However, all of these have been discussed in previous arguments for independent Claim 1 and have been proved that Anderson failed to suggest a user from a single web-browser UI to initiate multiple simultaneous concurrent tasks on targeted systems of either control system or system units.

In addition, the citation from [0029]-[0030] of Anderson also can not support the assertion of Office Action, where Anderson states that

“[0029] Turning first to the system configuration depicted under callout 4, we begin with a server 10. The server 10 is typically a web server and can run on a commercially available computer, such as a Sun Microsystems Enterprise Server.TM., available from Sun Microsystems in Mountain View, Calif., or a Dell.TM. or Gateway.TM. branded internet or application server. Such a system will include one or more microprocessors, a volatile memory area, a persistent memory area, and one or more mass storage devices. One or more sections of computer program code, or software, either in a compiled or an interpreted form, will run, for instance, in one of the memory areas, to cause the

microprocessor(s) to perform the sequences of operations and techniques described below. [0030] The server 10 should include a communications software stack 12, such as an IP (internet protocol) stack, and should be able to handle hypertext transfer protocol (HTTP) requests, secure socket layer (SSL) transactions, as well as a form of a standard generalized markup language (SGML), such as extensible markup language (XML), wireless markup language (WML), and optionally voice extensible markup language (VXML). Preferably, the variant of XML employed on the server is Microsoft's SOAP.TM. (Simple Object Access Protocol), although Java.TM. or X Windows.TM. could alternatively be employed. Hypertext markup language (HTML) files are preferably included on the server 10. The communications software stack 12 and the programming languages mentioned above are generally known in the art of network communications and interface design and are widely available."

A simple fact in computer industry is that none of web server, web-protocols (SOAP, HTTP, IP stack) and programming languages such as XML, WML, HTML have solved the issues of multiple concurrent tasks support within a single web-browser. Respectfully, these citations have failed to support the assertion of the Office Action. Accordingly, Anderson does not and can not teach or suggest Applicant's independent claim 37 either. Consequently, the Office Action's rejection of independent claim 37 should be withdrawn.

In view of the above, Applicant's independent claim 1 and 37 distinguish patentability over the Anderson and should be allowed. If the independent claims are valid, the claims that depend from the independent claims should also be valid as matter of law. See Jenric/Pentron, Inc. v. Dillon Co., 205 F. 3d 13377, 1382 (Fed. Cir.2000). Since claims 2, 3, 4, 5-6, 7-8, and 9-15 are depend on allowable independent Claim 1, Applicant respectfully submit that claims 2, 3, 4, 5-6, 7-8, and 9-15 should also be patentable as matter of law.

Rejections Under 35 USC 103(a)

In Paragraphs 23, Office Action has rejected claim 4 and 7-8 under 35 U.S.C. 103(a) as being un-patentable over Anderson (US 2003/0084128 AL) in view of Zhao (US 6,035,404). Applicant respectfully traverses the rejection and disagrees with Office Action assertion of claims 4, and 7-8 in view of following remarks.

Pursuant to MPEP § 2143.03:

2143.03 All Claim Limitations Must Be Taught or Suggested

To establish prima facie obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. In re Royka, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). "All words in a claim must be considered in judging the patentability of that claim against the prior art." In re Wilson, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970). If an independent claim is nonobvious under 35 U.S.C. 103, then any claim depending therefrom is nonobvious. In re Fine, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) (emphasis added).

Thus, under MPEP § 2143.03 and well-settled case law, where the rejection is based on a combination of references, the claim limitations must be taught or suggested by at least one of the references. Consequently, Applicant can show non-obviousness by attacking references individually where the rejection is based on a combination of references.

Applicant's independent claims are 1 and 37. None of these claims have been rejected on the ground of obviousness by the Office Action under 35 U.S.C. 103(a). Additionally, for the same reason discussed before, Anderson failed to suggest on independent claim 1 and 37 of Applicant. It shall be noticed that Claim 4, and 7-8 are dependent claims for Independent Claim 1. Consequently, the Office Action has failed to establish prima facie obviousness of dependent claim 4 and 7-8 because the elements and limits of claim 1 are not taught or suggested by Anderson. Based on this alone, the rejection on Claim 4, and 7-8 of Office Action must be withdrawn.

In addition, the claim 7 and 8 of present invention have been rejected for the similar reasons to the rejection of claim 4. The Office Action asserts (in part) that "As to claim 4, Anderson's invention can having multiple users logging in with their respective username/password on various console hosts. Anderson is silent on whether they can login concurrently. Zhao teaches multiple users logging in said console host through multiple web-consoles of multiple different console host concurrently (col. 7, lines 15-32, see Abstract) As to claim 7-8, they are rejected for similar reasons to the rejection of claim 4....". What Zhao col.7, line 15-32 says are:

"If the user is found to be a concurrent user in block 62, the system next determines from the mask data if the maximum number of simultaneous logins are existing at the present time (block 78). If not, 1 is added to the user login map (block 80), the login is permitted , and flow control proceeds to block 68 for

additional processing. If the maximum number of logins already exists, then the state lookup table (SLT) is consulted to determine which session for the same concurrent user IUID has been established for the longest time. That session is then given a termination or "time out" time (block 82) and that information is entered into the SLT. Flow control then proceeds to block 68 for additional processing. As in the case of the single user forced login situation, the concurrent user here may be advised of the time left for the current session to be timed out, thereby indicating to the logging in user the maximum time he needs to wait until he can get access to the system.”.

The Applicant disagrees with the assertion made by Office Action due to two reasons: First, from col.7, line 15-32 of Zhao, there is no any evidence shows that the control “mask” of Zhao can control login user to select a target system from multiple targeted systems on stateless network, but rather to control & limit login user to access a restricted system on stateless network (See first sentence in Abstract of Zhao that “System and method for managing user logins to a restricted computer service over a stateless network”, and See claim 1- claim 8 of Zhao). Opposite to Anderson, the present invention control login user to access one or more targeted systems in stateless network.

Second, as quoted and explored before that , login users of Anderson cannot run tasks on control management station and cannot select and run tasks on any targeted systems but only on a pre-configured system such as local computer of home system while present invention has no such limits, which has been described in claim 7-8 of present invention and demonstrated in following:

- 1) in paragraph 2 of page 6 of Applicant’s specification
- 2) in paragraph 1 of page 13 of Applicant, paragraph 1 of page 14 of present invent
- 3) in paragraph 2 & 3 on page 14 of present invention
- 4) in Fig. 1 and Fig 2 of present invention.

For example, the paragraph 2 of page 6 in specification of present invention states “With this invention, the multi-tasks support on the web-console in a simple environment (Fig 1) has been viewed as a special case of such support in a CCDSVM environment (Fig. 2). The CCDSVM (Fig. 2) will be degenerated into a

simple server environment (Fig. 1) if multiple system units (3 of Fig. 2) do not present”.

The paragraph 1 of page 13 in specification of present invention states “So that user A has obtained necessary information about all system units (3 of Fig. 2) and control management station (2 of Fig. 2) from console supporting software (6 of Fig 2). When user A initiates a task for a selected target system, which is either a system unit (3 of Fig. 2) or the control management station (2 of Fig. 2),”

Clearly, the combined references from Anderson and Zhao still failed to suggest how to control login user run tasks over control system or over one or multiple system units by providing user to select a target system from targeted systems. Again, the Office Action has failed to establish *prima facie* of obviousness for claim 4 and claim 7 - 8 because the limitation of claim 4, claim 7- 8 also can not be taught or suggested by combination of Anderson and Zhao. Consequently, the Office Action’s rejection of dependent claims 4, and claim 7-8 based on obviousness must be withdrawn.

In Paragraphs 26, Office Action also has rejected Claim 28-29 under 35 U.S.C. 103(a) as being un-patentable over Anderson (US 2003/0084128 AL) in view of Jackson (US 2002/0152305). The Office Action further asserts that (in part) “.... Anderson is silent in having a first level of security authentication for the control management station and a second level of security for the system units. However, Jackson teaches having and specifying security levels (with username/passwords) fro a plurality of processing engines with the capability of define logical volumes such as size....”.

Applicant respectfully traverses the rejection and disagrees with Office Action assertion on Claim 28-29 with following view: What [0400] of Jackson described is

“[0400] In addition to selectively interconnecting particular first processing engine/s 2100 to particular second processing engine/s 2120 using one or more distributed interconnect/s 2110, it is also possible to manage operations of an application specific buffer/cache 2122 and/or a file system specific logical volume manager 2124 over one or more distributed interconnect/s 2110 via a separate reserved or dedicated communication path across the distributed interconnect 2110. In this regard, operations of a buffer/cache 2122 that may be so managed include, but are not limited to, configuration, gathering performance

characteristics or data (e.g., to verify proper functioning), specifying security levels/gates (e.g., passwords, etc.). Operations of a logical volume manager 2124 that may be so managed include, but are not limited to, configuration (e.g., defining logical volumes and/or characteristics of logical volumes, such as defining number of RAID mirrors and size), loading content on to the logical volume manager (e.g., without interfering with user-access to data), etc. Advantageously, this embodiment may be used to provide a separate or reserved communication path for buffer/cache and/or logical volume manager management communication between a first processing engine 2100 and a second processing engine 2120 (e.g., inter-processor command communication between an application processing engine and a storage processing engine) that does not interfere with or reduce bandwidth for data/information (e.g., content) exchange between the first processing engine 2100 and the second processing engine 2120.” (Emphasis added).

Unfortunately, the [0400] and rest of specification of Jackson only mentioned security levels with username/password, but never suggest or explain in details on what are the other security levels are, which has been explicitly suggested as elements of this present invention with the first level and the second level security in claim 28-29. Again let us recall that “The identical invention must be shown in as complete details as is contained in the claim.” Richardson v Suzuki Motor Co., 869F.2d 1226,1236, 9USPQ2d 1913, 1920 (Fed. Cir. 1989) (See also MP.E.P 2131). Therefore, both Anderson and Jackson failed to suggest the limits of Claim 28-29. Further, both Anderson and Jackson failed to suggest the tasks of claim 28-29 of present invention, which can be executed at first level and second level of present invention as disclosed in first paragraph of page 15. Again, the Office Action has failed to establish *prima facie* of obviousness for Claim 28-29 because even combined Anderson and Jackson still failed to suggest the detailed first level and second level security of present invention. Therefore, the rejection on claim 28-29 based on obviousness also must be withdrawn.

Conclusion

Based on the foregoing remarks, Applicant clearly demonstrated that the present invention provides user from single web-browser to initiate multiple concurrent tasks for systems in CCDSVM environment while combined Anderson, Zhao, and Jackson failed to suggest or disclose such technology and methods. Therefore, Applicant believes that the rejections in the Office Action of May 21, 2007 are fully overcome, and that the

application is in condition for allowance. The issuance of a formal Notice of Allowance at an early date is requested.

Applicant thanks the Examiner for carefully examining the present application. If there are any questions, the Examiner is invited to contact Applicant at (408) 813-0536.

Date: 8/6/2007

By:


Sheng (Ted) Tai Tsao

Sheng (Ted) Tai Tsao
2979 Heidi Drive,
San Jose, CA 95132
Tel. 408-813-0536